# NN Impedance Control of Robots Interacting with Environments

Yanan Li[1], Shuzhi Sam Ge[1,2]*, Qun Zhang[1], Tong Heng Lee[3]

[1] Social Robotics Laboratory, Interactive Digital Media Institute, and NUS Graduate School for Integrative Sciences and Engineering, National University of Singapore, Singapore 119613

[2] Robotics Institute, and School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China

[3] Department of Electrical and Computer Engineering, National University of Singapore, Singapore 119077

**Abstract:** In this paper, neural networks impedance control is proposed for robot-environment interaction. Iterative learning control is developed to make the robot dynamics follow a given target impedance model. To cope with the problem of unknown robot dynamics, neural networks are employed such that neither the robot structure nor the physical parameters are required for the control design. The stability and performance of the resulted closed-loop system are discussed through rigorous analysis and extensive remarks. The validity and feasibility of the proposed method are verified through simulation studies.

*Index Terms* – impedance control, learning control, neural networks.

## 1 Introduction

Intelligent robots are envisioned not only to co-exist but also to collaborate and co-work with human beings in the foreseeable future for productivity, service, and operations with guaranteed quality. In these applications, a robot which is tightly controlled in position will face lots of challenges when it interacts with unknown environments. Under position control, the interaction force is deemed as disturbance and will be suppressed, which leads to a larger

*To whom all correspondences should be addressed.
Tel: (+65) 6516 6821, Fax: (+65) 6779 1103, E-mail: samge@nus.edu.sg

interaction force and may result in saturation, instability, and physical failure [1]. In the literature, there are two approaches which are able to assure compliant motion of robots. The first is hybrid position/force control [2, 3, 4, 5], which is aimed at controlling force and position in a nonconflicting way. The second is impedance control [6], which is aimed at developing a relationship between the contact force and position instead of controlling force or position separately. By specifying the relationship between the contact force and position, impedance control ensures that the robot is able to maneuver in a constrained environment while maintaining appropriate contact force [1].

While impedance control is acknowledged to be a promising way for a robot interacting with unknown environments, it has been studied and developed in many research works, such as [7, 8, 9, 10]. As uncertainties and complexities keep multiplying, one critical problem is impedance control design subject to unknown and uncertain robot dynamics. There have been extensive research works on adaptive impedance control carried out to cope with this problem in the literature. In [11], adaptive impedance control is developed to make the actual position of robot manipulators track the virtual desired trajectory, subject to uncertain robot parameters. As in most adaptive control methods including [11], the regressor introduced in [12] is needed and thus the robot structure is required to be known. In [13], function approximation technique is employed to approximate unknown and uncertain robot dynamics, and regressor-free adaptive impedance control is developed.

In parallel with adaptive impedance control, there has also been research effort on learning impedance control. In [14, 15], iterative learning impedance control is proposed where a sufficient condition to guarantee the learning convergence is required. In [16], an auxiliary error variable is introduced such that it is possible to extend existing methods in position control to impedance control. Based on the boundedness property, learning impedance control which requires neither the robot structure nor the physical parameters is developed in [16]. As further discussed in [17], if the bounds of the robot dynamics are known, the learning process is avoided while the high-gain scheme can be adopted. Compared to that in [14, 15], the approach developed in [16, 17] has a straightforward framework and fewer open parameters. It is thus more feasible in practical implementations and employed as the basis of the work in

this paper. In the method described in [16, 17], it is found that the high-gain feedback and the use of sign function are required, and the chattering exists when the defined impedance error becomes very small. Even if the sign function can be replaced by a smooth threshold function, the high-gain feedback still exists. In this regard, a model-based method is anticipated but it is contradictory to the fact that it is too tedious to obtain a robot model, as mentioned above. Similarly as in [13], a universal approximator can be employed to resolve this problem, and neural networks (NN) are thus considered in this paper. It has been demonstrated that NN are particularly suitable for controlling highly uncertain, nonlinear and complex systems, due to their excellent universal approximation ability to unknown complicated nonlinearities [18, 19, 20, 21]. The method using NN to approximate robot dynamics has been studied in the literature [22], which motivates the control design in this work.

Based on the above discussion, we investigate the problem of a robot interacting with unknown environments and develop NN impedance control. The method to be discussed in this paper is based on the learning mechanism as proposed in [16, 17], while NN are employed to cope with the problem of unknown robot dynamics. While the robot dynamics are not required in the learning impedance control to be developed in this paper, the adoption of the boundedness property in [16, 17] is also avoided. Then the high-gain feedback which is inherently along with the method in [16, 17] can be resolved. This will be illustrated in details through rigorous analysis and comparative simulation studies. Furthermore, it will be shown that the developed method guarantees compliant motion when a robot arm interacts with unknown environments and smooth transition between contact-free and contact phases.

Based on the above discussion, we highlight the contributions of this paper as follows:

(i) NN are employed to cope with the problem of unknown robot dynamics, such that neither the robot structure nor the physical parameters are required in the control design.

(ii) Learning control is developed based on NN approximation, and the use of the high-gain feedback in [16, 17] is avoided.

(iii) The defined impedance error is guaranteed to iteratively go to zero, while all the other signals in the closed-loop system are bounded.

The rest of the paper is organized as follows. In Section 2, robot dynamics and control objective are discussed. In Section 3, the details of the proposed learning control are presented, followed by the rigorous analysis. In Section 4, the validity of the proposed method is verified by simulation studies. Concluding remarks are given in Section 5.

# 2 System Overview

## 2.1 Robot Dynamics

The dynamics of the robot arm are described as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau - f(t) \tag{1}$$

where $M(q) \in \mathbb{R}^{n\times n}$ is the symmetric bounded positive definite inertia matrix; $C(q,\dot{q})\dot{q} \in \mathbb{R}^n$ denotes the Coriolis and Centrifugal force; $G(q) \in \mathbb{R}^n$ is the gravitational force; $\tau \in \mathbb{R}^n$ is the vector of control input; and $f(t) \in \mathbb{R}^n$ denotes the vector of interaction force exerted by the environment.

**Assumption 1** *It is assumed that the accurate force measurement is not achievable, i.e., there exists force measurement noise $\tilde{f} = \hat{f} - f \neq 0$, where $\hat{f}$ is the measurement of $f$. The force measurement noise is bounded with an unknown bound $b_f$, i.e., $\| \tilde{f} \| \leq b_f$.*

**Property 1** *[22] Matrix $M(q)$ is symmetric and positive definite.*

**Property 2** *[22] Matrix $2C(q,\dot{q}) - \dot{M}(q)$ is a skew-symmetric matrix if $C(q,\dot{q})$ is in the Christoffel form, i.e., $\xi^T(2C(q,\dot{q}) - \dot{M}(q))\xi = 0,\ \forall \xi \in \mathbb{R}^n$.*

As discussed in [23], the robot dynamics can be approximated by NN. Denote the elements of $M(q), C(q,\dot{q})$ and $G(q)$ as $m_{ij}(q), c_{ij}(q,\dot{q})$ and $g_i(q)$ for $i = 1,2,\ldots,n, j = 1,2,\ldots,n$,

respectively. Then, they are represented by

$$\begin{aligned} m_{ij}(q) &= \theta_{Mij}^T \xi_{Mij}(q) + \epsilon_{Mij} \\ c_{ij}(q,\dot{q}) &= \theta_{Cij}^T \xi_{Cij}(q,\dot{q}) + \epsilon_{Cij} \\ g_i(q) &= \theta_{Gi}^T \xi_{Gi}(q) + \epsilon_{Gi} \end{aligned} \tag{2}$$

where $\epsilon_{Mij}, \epsilon_{Cij}$ and $\epsilon_{Gi}$ are the bounded approximation errors, $\theta_{Mij}^T, \theta_{Cij}^T$ and $\theta_{Gi}^T$ are the column vectors of the NN weights, $\xi_{Mij}(q), \xi_{Cij}(q,\dot{q})$ and $\xi_{Gi}(q)$ are the vectors of Gaussian functions (activation functions) with elements

$$\begin{aligned} \xi_{Mijl}(q) &= \exp(\frac{-(q-\mu_{Ml})^T(q-\mu_{Ml})}{\sigma_M^2}) \\ \xi_{Cijl}(q) &= \exp(\frac{-(\eta-\mu_{Cl})^T(\eta-\mu_{Cl})}{\sigma_C^2}) \\ \xi_{Gil}(q) &= \exp(\frac{-(q-\mu_{Gl})^T(q-\mu_{Gl})}{\sigma_G^2}) \end{aligned} \tag{3}$$

where $l = 1, 2, \ldots, p$ and $p$ is the number of NN nodes, $\mu_{Ml}, \mu_{Cl}$ and $\mu_{Gl}$ are the centers of the functions, and $\sigma_M^2, \sigma_C^2$ and $\sigma_G^2$ are the variances, and $\eta = [q^T, \dot{q}^T]^T$.

**Remark 1** (GL matrices and operation [22]) *As the complexity and nonlinearity of individual entries of a matrix (vector) are different, to achieve roughly the same level of approximation accuracy, the sizes of the corresponding NN should also be different. The introduction of General-Leeway/Ge-Lee (GL) matrices* $\{*\}$ *and operation "*$\bullet$*" makes convenient expression and efficient computation possible for any general matrices/vectors in a manner with extra flexibility and leeway.*

*Suppose that there are three matrices* $A = [a_{ij}]$, $B = [b_{ij}]$ *and* $C = [c_{ij}]$, *where the elements* $a_{ij}$ *and* $b_{ij}$ *are column vectors, and* $c_{ij}$ *are scalars. The corresponding GL matrices have the following properties:*

$$\{A\}^T = [a_{ij}^T],\ \{A\}^T \bullet \{B\} = [a_{ij}^T b_{ij}],\ \{A\} \bullet C = [c_{ij} a_{ij}] \tag{4}$$

*Note that* $a_{ij}$ *and* $b_{ij}$ *may have different sizes for different* $i$ *and* $j$, *which increases the design*

*freedom and analysis efficiency [22].*

By employing NN and GL denotation, the robot dynamics are described as

$$\begin{aligned} M(q) &= \{\Theta_M\}^T \bullet \{\Xi_M(q)\} + E_M \\ C(q,\dot{q}) &= \{\Theta_C\}^T \bullet \{\Xi_C(q,\dot{q})\} + E_C \\ G(q) &= \{\Theta_G\}^T \bullet \{\Xi_G(q)\} + E_G \end{aligned} \tag{5}$$

where $\Theta_M, \Theta_C$ and $\Theta_G$ are matrices formed by $\theta_{Mij}, \theta_{Cij}$ and $\theta_{Gij}$, respectively, $\Xi_M(q), \Xi_C(q,\dot{q})$ and $\Xi_G(q)$ are matrices formed by $\xi_{Mij}(q), \xi_{Cij}(q,\dot{q})$ and $\xi_{Gij}(q)$, respectively, and $E_M, E_C$ and $E_G$ are matrices formed by $\epsilon_{Mij}, \epsilon_{Cij}$ and $\epsilon_{Gij}$, respectively. Because $E_M, E_C$ and $E_G$ are bounded, we denote their upper bounds as $b_M, b_C$ and $b_G$, respectively. Equivalently, we have

$$\|E_M\| \leq b_M,\ \|E_C\| \leq b_C,\ \|E_G\| \leq b_G \tag{6}$$

Note that $b_M, b_C$ and $b_G$ are unknown.

## 2.2 Impedance Control

As discussed in the Introduction, impedance control can be employed for a robot interacting with unknown environments. The stability of the coupled interaction system is guaranteed if the environments are passive.

Suppose that there is a desired impedance model given in the joint space

$$M_d\ddot{e} + C_d\dot{e} + G_d e = f \tag{7}$$

where $e = q_d - q$ with $q_d$ as the desired trajectory, and $M_d,\ C_d$ and $G_d$ are the desired inertia, damping and stiffness matrices, respectively.

The control objective in this paper is to find a sequence of control torques such that the impedance of the whole system tracks the given target impedance model (7). Before the control design, we need to construct an error signal between the real system and the virtual

system (7). The following impedance error in [14] is used

$$w = M_d\ddot{e} + C_d\dot{e} + G_d e - f \tag{8}$$

Then, the learning impedance control objective becomes

$$\lim_{k\to\infty} w^k(t) = 0, \quad \forall t \in [0, t_f] \tag{9}$$

where $k$ denotes the iteration number and $t_f$ is the iteration period. The problem under study is very difficult to solve by conventional control methods because we do not have complete knowledge of the robot arm. The situation becomes even more difficult when the unknown system parameters are time-varying due to payload changes, mechanical wear and so on. To overcome this difficulty, iterative learning control is proposed in the following, which searches for a desired control input through a sequence of repetitive operations with pre-specified operating conditions.

# 3 NN Impedance Control

In this section, NN impedance control is developed to achieve the control objective discussed in the above section. While the same framework as discussed in [16, 17] is adopted, some definitions and denotations are briefly introduced herein to make this paper self-contained.

For the analysis convenience, we define an augmented impedance error

$$\bar{w}^k = K_f w^k = \ddot{e}^k + K_d\dot{e}^k + K_p e^k - K_f f^k \tag{10}$$

where $K_d = M_d^{-1}C_d$, $K_p = M_d^{-1}G_d$ and $K_f = M_d^{-1}$.

By choosing two positive definite matrices $\Lambda$ and $\Gamma$ such that

$$\Lambda + \Gamma = K_d \text{ and } \dot{\Lambda} + \Gamma\Lambda = K_p \tag{11}$$

we further rewrite the augmented impedance error as

$$\bar{w}^k = \ddot{e}^k + (\Lambda + \Gamma)\dot{e}^k + (\dot{\Lambda} + \Gamma\Lambda)e^k - \dot{f}_l^k - \Gamma f_l^k \tag{12}$$

where $f_l^k$ satisfies

$$\dot{f}_l^k + \Gamma f_l^k = K_f f^k \tag{13}$$

**Remark 2** *In practical implementations, $M_d$, $C_d$ and $G_d$ are usually chosen to be diagonal matrices with constant elements. In this way, we have $\dot{\Lambda} = 0$, and $\Lambda$ and $\Gamma$ are also diagonal matrices that can be easily obtained according to (11).*

By defining

$$z^k = \dot{e}^k + \Lambda e^k - f_l^k \tag{14}$$

we obtain

$$\bar{w}^k = \dot{z}^k + \Gamma z^k \tag{15}$$

Suppose that $\lim_{k\to\infty} \dot{z}^k$ exists, $\lim_{k\to\infty} z^k = 0$ will lead to $\lim_{k\to\infty} \dot{z}^k = 0$, and thus $\lim_{k\to\infty} w^k = 0$ considering (15) and (10). Based on this fact, the control objective becomes $z^k \to 0$ as $k \to \infty$.

Let the estimates of $M(q), C(q,\dot{q})$ and $G(q)$ be $\hat{M}(q), \hat{C}(q,\dot{q})$ and $\hat{G}(q)$, respectively, and they are defined as

$$\begin{aligned} \hat{M}(q) &= \{\hat{\Theta}_M\}^T \bullet \{\Xi_M(q)\} \\ \hat{C}(q,\dot{q}) &= \{\hat{\Theta}_C\}^T \bullet \{\Xi_C(q,\dot{q})\} \\ \hat{G}(q) &= \{\hat{\Theta}_G\}^T \bullet \{\Xi_G(q)\} \end{aligned} \tag{16}$$

where $\hat{\Theta}_M, \hat{\Theta}_C$ and $\hat{\Theta}_G$ are the estimates of $\Theta_M, \Theta_C$ and $\Theta_G$, respectively.

The control input is proposed as

$$\tau^k = \tau_{ct}^k + \tau_{fb}^k + \tau_{com}^k + \hat{f}^k \tag{17}$$

where $\tau_{ct}^k$, $\tau_{fb}^k$ and $\tau_{com}^k$ are the computed torque vector, feedback torque vector and compensation torque vector, respectively.

In specific, the computed torque vector is given by

$$\tau_{ct}^k = \hat{M}^k(q)\ddot{q}_r^k + \hat{C}^k(q,\dot{q})\dot{q}_r^k + \hat{G}^k(q) \tag{18}$$

where

$$\begin{aligned} \dot{q}_r^k &= \dot{q}_d + \Lambda e^k - \hat{f}_l^k \\ \ddot{q}_r^k &= \ddot{q}_d + \Lambda \dot{e}^k - \dot{\hat{f}}_l^k \end{aligned} \tag{19}$$

with $\hat{f}_l^k$ satisfying $\dot{\hat{f}}_l^k + \Gamma \hat{f}_l^k = K_f \hat{f}^k$.

By defining

$$\bar{z}^k = \dot{e}^k + \Lambda e^k - \hat{f}_l^k = z^k - \tilde{f}_l^k \tag{20}$$

with $\tilde{f}_l^k = \hat{f}_l^k - f_l^k$, the compensation torque vector is given by

$$\tau_{com}^k = -L^k \hat{B}^k \tag{21}$$

where $L^k = [\mathrm{sgn}(\bar{z}^k), \mathrm{sgn}(\bar{z}^k)\|\ddot{q}_r^k\|, \mathrm{sgn}(\bar{z}^k)\|\dot{q}_r^k\|]$ and $\hat{B}$ is the estimate of $B = [b_f + b_G, b_M, b_C]^T$. The following analysis will show that the compensation torque vector (21) will compensate for not only the inaccurate force measurement, but also the NN estimation error.

The feedback torque vector is given by

$$\tau_{fb}^k = -K\bar{z}^k \tag{22}$$

where $K$ is a symmetric positive definite matrix.

To obtain $\hat{M}^k(q), \hat{C}^k(q,\dot{q})$ and $\hat{G}^k(q)$ in (18) and $\hat{B}^k$ in (21), we develop the following learning law

$$\begin{aligned}
\hat{\Theta}_M^k &= \hat{\Theta}_M^{k-1} - S_M^{-1} \bullet \{\Xi_M(q)\}\bar{z}^k \ddot{q}_r^{k^T} \\
\hat{\Theta}_C^k &= \hat{\Theta}_C^{k-1} - S_C^{-1} \bullet \{\Xi_C(q,\dot{q})\}\bar{z}^k \dot{q}_r^{k^T} \\
\hat{\Theta}_G^k &= \hat{\Theta}_G^{k-1} - S_G^{-1}\{\Xi_G(q)\} \bullet \bar{z}^k \\
\hat{B}^k &= \hat{B}^{k-1} + S_B^{-1} L^{k^T} \bar{z}^k
\end{aligned} \tag{23}$$

where $S_M, S_C, S_G$ and $S_B$ are symmetric positive definite matrices, and $\hat{\Theta}_M^k, \hat{\Theta}_C^k, \hat{\Theta}_G^k$ and $\hat{B}^k$ are the estimates of $\Theta_M^k, \Theta_C^k, \Theta_G^k$ and $B^k$, respectively.

**Remark 3** *The variables in the control input (17) include $\dot{q}_d$, $\ddot{q}_d$, $e$, $\dot{e}$, $q$, $\dot{q}$, $\hat{f}_l$ and $\dot{\hat{f}}_l$, which are all available. Note that $\hat{f}_l$ and $\dot{\hat{f}}_l$ are calculated from the measured force $\hat{f}$, of which the derivative is not needed.*

Substituting the control input (17) with (18), (21) and (22) into the dynamics (1), we obtain the closed-loop dynamics as below

$$M^k(q)\dot{\bar{z}}^k + C^k(q,\dot{q})\bar{z}^k = -(\tilde{M}^k(q)\ddot{q}_r^k + \tilde{C}^k(q,\dot{q})\dot{q}_r^k + \tilde{G}^k(q) + K\bar{z}^k + L^k\hat{B}^k + \tilde{f}^k) \tag{24}$$

where $\tilde{M}^k(q) = M^k(q) - \hat{M}^k(q)$, $\tilde{C}^k(q,\dot{q}) = C^k(q,\dot{q}) - \hat{C}^k(q,\dot{q})$ and $\tilde{G}^k(q) = G^k(q) - \hat{G}^k(q)$. Note that we have the following equations

$$\begin{aligned}
\tilde{M}^k(q) &= \{\tilde{\Theta}_M^k\}^T \bullet \{\Xi_M^k(q)\} + E_M^k \\
\tilde{C}^k(q,\dot{q}) &= \{\tilde{\Theta}_C^k\}^T \bullet \{\Xi_C^k(q,\dot{q})\} + E_C^k \\
\tilde{G}^k(q) &= \{\tilde{\Theta}_G^k\}^T \bullet \{\Xi_G^k(q)\} + E_G^k
\end{aligned} \tag{25}$$

where $\tilde{\Theta}_M^k = \hat{\Theta}_M^k - \Theta_M, \tilde{\Theta}_C^k = \hat{\Theta}_C^k - \Theta_C, \tilde{\Theta}_G^k = \hat{\Theta}_G^k - \Theta_G$ and $\tilde{B}^k = \hat{B}^k - B$.

**Theorem 1** *Considering the system described by (1) under Assumption 1, with the control input (17) and the learning law (23), we have the following results:*

*(i)* $\lim_{k\to\infty} w^k(t)$ *is bounded by* $\|M_d\Gamma\|b_f$ *for all* $t \in [0, t_f]$, *i.e.,* $\|\lim_{k\to\infty} w^k(t)\| \leq b_f$. *When the force measurement is accurate,* $b_f = 0$ *indicates* $\lim_{k\to\infty} w^k(t) = 0$.

*(ii) all the signals in the closed-loop are bounded for all* $t \geq 0$.

**Proof 1** *Consider the following Lyapunov-Krasovskii functional*

$$\Omega^k(t) = U^k(t) + V^k(t) + W^k(t) \tag{26}$$

*where*

$$\begin{aligned} U^k(t) &= \frac{1}{2}\bar{z}^{k^T} M^k \bar{z}^k \\ V^k(t) &= \frac{1}{2}\int_0^t [tr(\tilde{\Theta}_M^{k^T} S_M^T \tilde{\Theta}_M^k + \tilde{\Theta}_C^{k^T} S_C^T \tilde{\Theta}_C^k) + \tilde{\Theta}_G^{k^T} S_G^T \tilde{\Theta}_G^k] d\tau \\ W^k(t) &= \frac{1}{2}\int_0^t \tilde{B}^{k^T} S_B^T \tilde{B}^k d\tau \end{aligned} \tag{27}$$

*where* $tr(\cdot)$ *denotes the matrix trace.*

*According to Property 2 and closed-loop dynamics (24), we have*

$$\begin{aligned} U^k(t) &= U^k(0) + \int_0^t [\bar{z}^{k^T} M^k(q)\dot{\bar{z}}^k + \frac{1}{2}\bar{z}^{k^T}\dot{M}^k(q)\bar{z}^k] d\tau \\ &= U^k(0) + \int_0^t \bar{z}^{k^T}[M^k(q)\dot{\bar{z}}^k + C^k(q,\dot{q})\bar{z}^k] d\tau \\ &= U^k(0) - \int_0^t \bar{z}^{k^T}[\tilde{M}^k(q)\ddot{q}_r^k + \tilde{C}^k(q,\dot{q})\dot{q}_r^k + \tilde{G}^k(q) + K\bar{z}^k + L^k\hat{B}^k + \tilde{f}^k] d\tau \\ &= -\int_0^t \bar{z}^{k^T}[\tilde{M}^k(q)\ddot{q}_r^k + \tilde{C}^k(q,\dot{q})\dot{q}_r^k + \tilde{G}^k(q) + K\bar{z}^k + L^k\hat{B}^k + \tilde{f}^k] d\tau \end{aligned} \tag{28}$$

*where we use the assumption that* $U^k(0) = 0$. *This is obtained by assuming that* $\dot{q}^k(0) = \dot{q}_d(0)$, $q^k(0) = q_d(0)$ *and* $\hat{f}^k(0) = 0$, *which are known as the resetting condition [24].*

*Besides, we consider*

$$\begin{aligned} & V^k(t) - V^{k-1}(t) \\ = & -\int_0^t [tr(\frac{1}{2}\delta\tilde{\Theta}_M^{k^T} S_M \delta\tilde{\Theta}_M^k + \tilde{\Theta}_M^{k^T} S_M \delta\tilde{\Theta}_M^k + \frac{1}{2}\delta\tilde{\Theta}_C^{k^T} S_C \delta\tilde{\Theta}_C^k + \tilde{\Theta}_C^{k^T} S_C \delta\tilde{\Theta}_C^k) \\ & +\frac{1}{2}\delta\tilde{\Theta}_G^{k^T} S_G \delta\tilde{\Theta}_G^k + \tilde{\Theta}_G^{k^T} S_G \delta\tilde{\Theta}_G^k] d\tau \\ \leq & -\int_0^t [tr(\tilde{\Theta}_M^{k^T} S_M \delta\tilde{\Theta}_M^k + \tilde{\Theta}_C^{k^T} S_C \delta\tilde{\Theta}_C^k) + \tilde{\Theta}_G^{k^T} S_G \delta\tilde{\Theta}_G^k] d\tau \end{aligned} \tag{29}$$

*By defining* $\delta\tilde{\Theta}_M^k = \tilde{\Theta}_M^{k-1} - \tilde{\Theta}_M^k$, $\delta\tilde{\Theta}_C^k = \tilde{\Theta}_C^{k-1} - \tilde{\Theta}_C^k$ *and* $\delta\tilde{\Theta}_G^k = \tilde{\Theta}_G^{k-1} - \tilde{\Theta}_G^k$, *we obtain the following equations from (23)*

$$\begin{aligned} \delta\tilde{\Theta}_M^k &= -S_M^{-1} \bullet \{\Xi_M(q)\} \bar{z}^k \ddot{q}_r^{k^T} \\ \delta\tilde{\Theta}_C^k &= -S_C^{-1} \bullet \{\Xi_C(q,\dot{q})\} \bar{z}^k \dot{q}_r^{k^T} \\ \delta\tilde{\Theta}_G^k &= -S_G^{-1} \{\Xi_G(q)\} \bullet \bar{z}^k \end{aligned} \tag{30}$$

*Based on the above results, we have*

$$\begin{aligned} & -\int_0^t [tr(\tilde{\Theta}_M^{k^T} S_M \delta\tilde{\Theta}_M^k + \tilde{\Theta}_C^{k^T} S_C \delta\tilde{\Theta}_C^k) + \tilde{\Theta}_G^{k^T} S_G \delta\tilde{\Theta}_G^k] d\tau \\ = & \int_0^t [tr[(\{\tilde{\Theta}_M^k\}^T \bullet \{\Xi_M(q)\})(\bar{z}^k \ddot{q}_r^{k^T}) + (\{\tilde{\Theta}_C^k\}^T \bullet \{\Xi_C(q,\dot{q})\})(\bar{z}^k \dot{q}_r^{k^T})] \\ & +\tilde{\Theta}_G^{k^T}(\{\Xi_G(q)\} \bullet \bar{z}^k)] d\tau \\ = & \int_0^t [tr[(\ddot{q}_r^k \bar{z}^{k^T})(\{\tilde{\Theta}_M^k\}^T \bullet \{\Xi_M(q)\}) + (\dot{q}_r^k \bar{z}^{k^T})(\{\tilde{\Theta}_C^k\}^T \bullet \{\Xi_C(q,\dot{q})\})] \\ & +\bar{z}^{k^T}(\{\tilde{\Theta}_G^k\}^T \bullet \{\Xi_G(q)\})] d\tau \\ = & \int_0^t [tr[\ddot{q}_r^k \bar{z}^{k^T}(\tilde{M}^k(q) - E_M^k) + \dot{q}_r^k \bar{z}^{k^T}(\tilde{C}^k(q,\dot{q}) - E_C^k)] \\ & +\bar{z}^{k^T}(\tilde{G}^k(q) - E_G^k)] d\tau \end{aligned} \tag{31}$$

*Considering the following fact*

$$\begin{aligned} tr[\ddot{q}_r^k \bar{z}^{k^T}(\tilde{M}^k(q) - E_M^k)] &= \bar{z}^{k^T}(\tilde{M}^k(q) - E_M^k)\ddot{q}_r^k \\ tr[\dot{q}_r^k \bar{z}^{k^T}(\tilde{C}^k(q,\dot{q}) - E_C^k)] &= \bar{z}^{k^T}(\tilde{C}^k(q,\dot{q}) - E_C^k)\dot{q}_r^k \end{aligned} \tag{32}$$

*we have*

$$\begin{aligned}
&\int_0^t [tr[\ddot{q}_r^k \bar{z}^{k^T}(\tilde{M}^k(q) - E_M^k) + \dot{q}_r^k \bar{z}^{k^T}(\tilde{C}^k(q,\dot{q}) - E_C^k)] \\
&+\bar{z}^{k^T}(\tilde{G}^k(q) - E_G^k)]d\tau \\
= &\int_0^t \bar{z}^{k^T}[(\tilde{M}^k(q) - E_M^k)\ddot{q}_r^k + (\tilde{C}^k(q,\dot{q}) - E_C^k)\dot{q}_r^k \\
&+(\tilde{G}^k(q) - E_G^k)]d\tau \\
= &\int_0^t \bar{z}^{k^T}(\tilde{M}^k(q)\ddot{q}_r^k + \tilde{C}^k(q,\dot{q})\dot{q}_r^k + \tilde{G}^k(q))d\tau \\
&-\int_0^t \bar{z}^{k^T}(E_M^k\ddot{q}_r^k + E_C^k\dot{q}_r^k + E_G^k)d\tau
\end{aligned} \tag{33}$$

*Considering (29), (31) and (33), we obtain*

$$\begin{aligned}
&V^k(t) - V^{k-1}(t) \\
\leq &\int_0^t \bar{z}^{k^T}(\tilde{M}^k(q)\ddot{q}_r^k + \tilde{C}^k(q,\dot{q})\dot{q}_r^k + \tilde{G}^k(q))d\tau \\
&-\int_0^t \bar{z}^{k^T}(E_M^k\ddot{q}_r^k + E_C^k\dot{q}_r^k + E_G^k)d\tau
\end{aligned} \tag{34}$$

*Furthermore, by defining* $\delta\tilde{B}^k = \tilde{B}^{k-1} - \tilde{B}^k$*, we have* $\delta\tilde{B}^k = -S_B^{-1}L^{k^T}\bar{z}^k$*, and*

$$\begin{aligned}
&W^k(t) - W^{k-1}(t) \\
= &\int_0^t (\frac{1}{2}\tilde{B}^{k^T}S_B^T\tilde{B}^k - \tilde{B}^{k-1^T}S_B^T\tilde{B}^{k-1})d\tau \\
= &-\int_0^t (\delta\tilde{B}^{k^T}S_B^T\tilde{B}^k + \frac{1}{2}\delta\tilde{B}^{k^T}S_B^T\delta\tilde{B}^k)d\tau \\
\leq &-\int_0^t \delta\tilde{B}^{k^T}S_B^T\tilde{B}^k d\tau = \int_0^t \bar{z}^{k^T}L^k\tilde{B}^k d\tau
\end{aligned} \tag{35}$$

*According to (26), (28), (34) and (35), we have the following result*

$$\begin{aligned}
\Delta\Omega^k(t) &= \Omega^k(t) - \Omega^{k-1}(t) \\
&= (U^k(t) - U^{k-1}(t)) + (V^k(t) - V^{k-1}(t)) + (W^k(t) - W^{k-1}(t)) \\
&\leq -\int_0^t (\bar{z}^{k^T}(K\bar{z}^k + L^k\hat{B}^k + \tilde{f}^k))d\tau \\
&\quad -\int_0^t \bar{z}^{k^T}(E_M^k\ddot{q}_r^k + E_C^k\dot{q}_r^k + E_G^k)d\tau + \int_0^t \bar{z}^{k^T}L^k\tilde{B}^k d\tau \\
&\leq -\int_0^t \bar{z}^{k^T}K\bar{z}^k d\tau - \int_0^t (\bar{z}^{k^T}(L^kB - L^kB))d\tau \\
&= -\int_0^t \bar{z}^{k^T}K\bar{z}^k d\tau
\end{aligned} \tag{36}$$

*In the above derivation, the following result is used*

$$\begin{aligned}
& -\bar{z}^{k^T}(\tilde{f}^k + E_M^k\ddot{q}_r^k + E_C^k\dot{q}_r^k + E_G^k) \\
\leq\ & \|\bar{z}^k\|(\|\tilde{f}^k\| + \|E_M^k\ddot{q}_r^k\| + \|E_C^k\dot{q}_r^k\| + \|E_G^k\|) \\
\leq\ & \|\bar{z}^k\|(\|\tilde{f}^k\| + \|E_M^k\|\|\ddot{q}_r^k\| + \|E_C^k\|\|\dot{q}_r^k\| + \|E_G^k\|) \\
\leq\ & \|\bar{z}^k\|(b_f + b_M\|\ddot{q}_r^k\| + b_C\|\dot{q}_r^k\| + b_G) \\
\leq\ & \bar{z}^{k^T}sgn(\bar{z}^k)((b_f + b_G) + b_M\|\ddot{q}_r^k\| + b_C\|\dot{q}_r^k\|) \\
=\ & \bar{z}^{k^T}L^kB^T
\end{aligned} \tag{37}$$

*Assuming that $\Omega^0$ is bounded for all $t \in [0, t_f]$, (36) indicates that the monotonically decreasing nonnegative sequence $\Omega^k$ converges to a nonnegative fixed value, thus we have $\Delta\Omega^k \to 0$ as $k \to \infty$.*

*Considering that*

$$\Delta\Omega^k \leq -\bar{z}^{k^T}K\bar{z}^k \leq 0 \tag{38}$$

*we obtain*

$$\lim_{k\to\infty} \bar{z}^k = 0 \tag{39}$$

*Considering the definition of $\bar{z}$ in (20), we obtain*

$$\lim_{k\to\infty} z^k = \lim_{k\to\infty} \tilde{f}_l^k \tag{40}$$

*It follows from (10), (15) and the above equation that*

$$\lim_{k\to\infty} w^k(t) = \lim_{k\to\infty} \tilde{f}^k(t) \tag{41}$$

*which immediately leads to*

$$\| \lim_{k\to\infty} w^k(t)\| \leq b_f \tag{42}$$

*When the force measurement is accurate, $b_f = 0$ indicates $\lim_{k\to\infty} w^k(t) = 0$. It completes the proof.*

**Remark 4** *As discussed in the Introduction, the linear-in-the-parameters property is considered in most adaptive/learning methods [12], and the regressor is used in the control design. However, the usage of the regressor indicates a requirement that the robot structure is known as* a priori *knowledge. The computation of the regressor is quite tedious especially when the robot arm has a high DOF. In [16], this problem has been investigated by employing the boundedness property, and a learning method was developed to "learn" unknown bounds $k_M,\ k_C,\ k_G$. It has also been shown that if the bounds $k_M,\ k_C,\ k_G$ are known, the learning process can be further avoided by employing the high-gain scheme [17]. In the control input proposed in [16, 17], the sign function and high-gain feedback are used which may cause chattering or even system instability during implementations. It can be partially solved by replacing the sign function by a smooth threshold function, but the high-gain feedback still exists. By employing NN, the unknown robot dynamics instead of the unknown bounds are estimated in this work, and thus the high-gain feedback is avoided in the computed torque component.*

**Remark 5** *Note that sign function still appears in the control input (17), but it is only used in (21) to compensate for inaccurate force measurement and NN estimation error. In other words, sign function is not necessarily to be employed in the proposed method, if the force measurement*

*and NN approximation are accurate enough or a prescribed small error is acceptable.*

**Remark 6** *Although NN are employed in the control design discussed in this section, it can be replaced by other linearly parameterized function approximators such as fuzzy systems, polynomials, splines etc.*

**Remark 7** *Instead of iterative learning, the method developed in this paper can be used to develop an adaptive NN control. While iterative learning requires the operations to be repeated over and over again, it provides an advantage over the adaptive control that the system parameters of the robot dynamics can be time-varying during a learning iteration. Therefore, iterative learning and adaptive control can be chosen according to different system setups.*

# 4 Simulation Studies

In this section, we conduct the simulation using the Robotics Toolbox introduced in [25]. A two-DOF robot arm with two revolute joints moves in the $X-Y$ plane, as shown in Fig. 1. The robot arm repeats its motion to track the desired trajectory in each iteration, and is repositioned to its initial position at the beginning of each iteration. In the following, $m_i, l_i, In_i$ and $l_{ci}$, $i = 1, 2$, represent the mass, the length, the inertia about the Z-axis that comes out of the page passing through the center of mass, and the distance from the previous joint to the center of mass of link $i$, respectively. And we set $m_1 = m_2 = 1.0$kg, $l_1 = l_2 = 0.2$m, $In_1 = In_2 = 0.01$kgm$^2$, and $l_{c1} = l_{c2} = 0.1$m. Note that these parameters are only used for the simulation and they will not be used in the control design. The initial position of the robot arm at the $k$th iteration is $q^k(0) = [-\frac{\pi}{3}, \frac{2\pi}{3}]^T$.

The desired trajectory of the robot arm in the Cartesian space is specified by

$$x_d(t) = 0.2 + 0.1(6t^5 - 15t^4 + 10t^3),\ y_d(t) = 0 \tag{43}$$

where $t \in [0, t_f]$ and $t_f = 1$s.

The desired impedance model is specified by (7) with

$$M_d = 0.1I_2,\ C_d = 8I_2 \text{ and } G_d = 8I_2 \tag{44}$$

where $I_2$ represents a $2 \times 2$ unit matrix.

Consider the control input (17) with each component (18), (21) and (22), and the updating law (23). In (18), we choose $\mu_{Ml} = 0.1$, $\mu_{Cl} = 0.1$, $\mu_{Gl} = 0.1$, $\delta_M = 1$, $\delta_C = 1$ and $\delta_G = 1$, for $l = 1, 2, \ldots, 10$. In (22), we choose $K = I_2$. And in (23), we choose $S_M = 0.33I_2$, $S_C = 0.25I_2$, $S_G = 0.33I_2$ and $S_B = 0.67I_2$ to obtain $\hat{\Theta}_M^k$, $\hat{\Theta}_C^k$, $\hat{\Theta}_G^k$ and $\hat{B}^k$. Similarly as in [16], no dynamics information is needed so the control design is straightforward and simple. While the above parameters do not guarantee the best control performance, it is feasible to change them with other values.

In the first case of this simulation, the robot arm is considered to be contact-free, which indicates that there is no external force exerted by the environment. The defined impedance error in the joint space and positions in $X$ and $Y$ directions are shown in Figs. 2, 3 and 4, which illustrate the results at $k = 1$, $k = 10$ and $k = 30$ respectively. It is easy to find that the impedance error becomes smaller as the iteration number increases. At $k = 30$, the impedance errors at two directions almost go to zero, as shown in Fig. 4, which indicates that the dynamics of the robot arm are governed by the desired impedance model. As there is no external force from the environment, the actual position converges to the desired trajectory, which can be found in the last two sub-figures of Figs. 2, 3 and 4. Theoretically, the learning process will not stop until $k \to \infty$. However, in the practical implementations, the learning process can be manually stopped when the impedance error falls into a pre-defined small set.

In the second case, it is considered that there is an external force exerted by the environment to the end-effector of the robot arm at the $X$ direction. The environment is described by $f_x = K_e(x - x_0)$ which is a model widely used in the literature, e.g., [26, 27]. $f_x$ is the interaction force in the Cartesian space and $x_0 = 0.2$ is the rest position of the environment. $K_e = 2(1+0.5\Delta)$ is the stiffness where $\Delta$ stands for the uncertainty and it is a pseudo-random value drawn from a uniform distribution on the unit interval. Note that this environment

model is only used for the simulation and it is unknown for the control design. The force measurement noise is a uniform-random-number signal with amplitude of 0.01. The defined impedance error and positions in $X$ and $Y$ directions at $k = 1$, $k = 10$ and $k = 30$ are shown in Figs. 5, 6 and 7, respectively. It is found that the positions in $X$ and $Y$ directions drift away from the desired trajectories due to the effect of the external force, which is different from that in the first case. Nevertheless, as the iteration number increases, the impedance error becomes smaller and converges to zero as the iteration number becomes very large. This is similar to that in the first case and indicates that the proposed method guarantees the robot dynamics governed by the desired impedance model in both contact-free and contact cases.

Furthermore, the above results may be achieved by learning control proposed in [16]. The method in [16] is based on a property that the robot dynamics are bounded and the high-gain feedback is required in the computed torque component, as discussed in Remark 4. For the comparison purpose, the results with learning control in [16] in two cases of contact-free and contact are shown in Figs. 8 and 9, respectively. The learning rate of the method in [16] is $L = 0.04I_2$ and other parameters are the same as in above simulation studies. Compared to that in Figs. 4 and 7, similar results of impedance error and trajectory are found in Figs. 8 and 9, but there exists an obvious chattering in both figures. In this regard, NN based method proposed in this paper is preferred. Nonetheless, it is also necessary to note that using NN based method increases complexity and thus reduces robustness compared to the method in [16]. Therefore, which one to be adopted in practical implementations needs to be evaluated by considering the computation complexity and possible sacrifice of system stability.

# 5 Conclusion

In this work, robot-environment interaction has been investigated. Learning control has been developed to make the robot dynamics governed by the given target impedance model. By adopting NN, neither the linear-in-parameters property nor the dynamics boundedness property was needed. The control performance has been discussed through rigorous proof and remarking arguments. The simulation results have shown the validity of the proposed method

and the superiority over the existing methods.

# References


[1] Kazerooni, H., and Sheridan, T. B., and Houpt, P. K.: 'Robust compliant motion for manipulators, part I: the fundamental concepts of compliant motion', IEEE Journal of Robotics and Automation, 1986, RA-2, (2), pp. 83-92

[2] Queiroz, M. D., and Hu, J., and Dawson, D., and Burg, T., and Donepudi, S.: 'Adaptive position/force control of robot manipulators without velocity measurements: theory and experimentation', IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics, 1997, 27, (5), pp. 796-809

[3] Dupree, K., and Liang, C. H., and Hu, G., and Dixon, W. E.: 'Adaptive Lyapunov-based control of a robot and mass-spring system undergoing an impact collision', IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics, 2008, 38, (4), pp. 1050-1061

[4] Doulgeri, Z., and Iliadis, G.: 'Stability of a contact task for a robotic arm modelled as a switched system', IET Control Theory and Applications, 2007, 1, (3), pp. 844-853

[5] Karayiannidis, Y., and Doulgeri, Z.: 'Blind force/position control on unknown planar surfaces', IET Control Theory and Applications, 2009, 3, (5), pp. 595-603

[6] Hogan, N.: 'Impedance control: an approach to manipulation-part I: theory; part II: implementation; part III: applications', Journal of Dynamic Systems, Measurement and Control, 1985, 107, (1), pp. 1-24

[7] Gillespie, R. B., and Colgate, J. E., and Peshkin, M. A.: 'A general framework for Cobot control', IEEE Transactions on Robotics and Automation, 2001, 17, (4), pp. 391-401

[8] Lynch, K. M., and Liu, C., and Sorensen, A., and Kim, S., and Peshkin, M., and Colgate, J. E., and Tickel, T., and Hannon, D., and Shiels, K.: 'Motion guides for assisted manipulation', The International Journal of Robotics Research, 2002, 21, (1), pp. 27-43

[9] Buerger, S. P., and Hogan, N.: 'Complementary stability and loop shaping for improved human-robot interaction', IEEE Transactions on Robotics, 2007, 23, (2), pp. 232-244

[10] Hirata, Y., and Wang, Z., and Fukaya, K., and Kosuge, K.: 'Transporting an object by a passive mobile robot with servo brakes in cooperation with a human', Advanced Robotics, 2009, 23, pp. 387-404

[11] Lu, W.-S., and Meng, Q.-H.: 'Impedance control with adaptation for robotic manipulations', IEEE Transactions on Robotics and Automation, 1991, 7, (3), pp. 408-415

[12] Slotine, J-J. E., and Li, W.: 'On the adaptive control of robotic manipulators', The International Journal of Robotics Research, 1987, 6, (3), pp. 49-59

[13] Chien, M. C., and Huang, A. C.: 'Adaptive impedance control of robot manipulators based on function approximation technique', Robotica, 2004, 22, pp. 395-403

[14] Wang, D., and Cheah, C. C.: 'An iterative learning-control scheme for impedance control of robotic manipulators', The International Journal of Robotics Research, 1998, 17, (10), pp. 1091-1099

[15] Cheah, C. C., and Wang, D.: 'Learning impedance control for robotic manipulators', IEEE Transactions on Robotics and Automation, 1998, 14, (3), pp. 452-465

[16] Li, Y., and Ge, S. S., and Yang, C.: 'Learning impedance control for physical robot-environment interaction', International Journal of Control, 2012, 85, (2), pp. 182-193

[17] Li, Y., and Ge, S. S., and Yang, C., and Li, X., and Tee, K. P.: 'Model-free impedance control for safe human-robot interaction', Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China, May 2011, pp. 6021-6026

[18] Chen, T. P., and Chen, H.: 'Approximation capability to functions of several variables, nonlinear functionals, and operators by radial basis function neural networks', IEEE Transactions on Neural Networks, 1995, 6, (4), pp. 904-910

[19] Cherkassky, V., and Ghering, D., and Mulier, F.: 'Comparison of adaptive methods for function estimation from samples', IEEE Transactions on Neural Networks, 1996, 7, (4), pp. 969-984

[20] Ge, S. S., and Ren, B., and Tee, K. P., and Lee, T. H.: 'Approximation based control of uncertain helicopter dynamics', IET Control Theory and Applications, 2009, 3, (7), pp. 941-956

[21] Park, S. H., and Han, S. I.: 'Robust-tracking control for robot manipulator with deadzone and friction using backstepping and RFNN controller', IET Control Theory and Applications, 2010, 5, (12), pp. 1397-1417

[22] Ge, S. S., and Lee, T. H., and Harris, C. J.: 'Adaptive neural network control of robotic manipulators'(World Scientific, 1998)

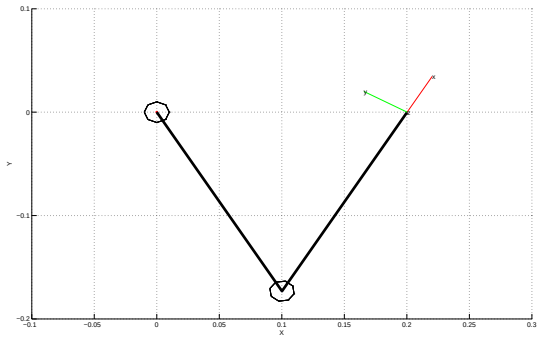
[23] Ge, S. S., and Hang, C. C., and Woon, L. C.: 'Adaptive neural network control of robot manipulators in task space', IEEE Transactions on Industrial Electronics, 1997, 44, (6), pp. 746-752

[24] Xu, J.-X., and Badrinath, V., and Qu, Z.: 'Robust learning control for robotic manipulators with an extension to a class of nonlinear systems', International Journal of Control, 2000, 73, (10), pp. 858-870

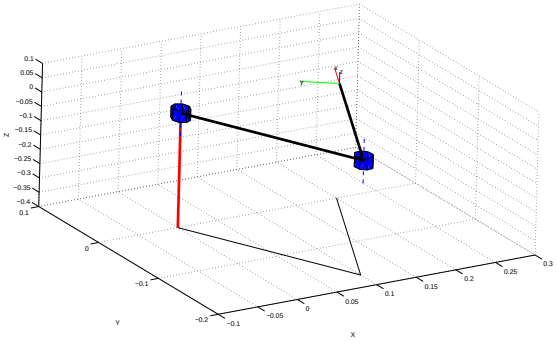
[25] Corke, P. I.: 'A robotics toolbox for Matlab', IEEE Robotics and Automation Magazine, 1996, 3, (1), pp. 24-32

[26] Seraji, H. and Colbaugh, R.: 'Force tracking in impedanc control', The International Journal of Robotics Research, 1997, 16, (1), pp. 97-117

[27] Jung, S., and Hsia, T. C., and Bonitz, R. G.: 'Force tracking impedance control for robot manipulators with an unknown environment: theory, simulation, and experiment', The International Journal of Robotics Research, 2001, 20, (9), pp. 765-774




(a) plane view




(b) stereo view

Figure 1: Simulation scenario

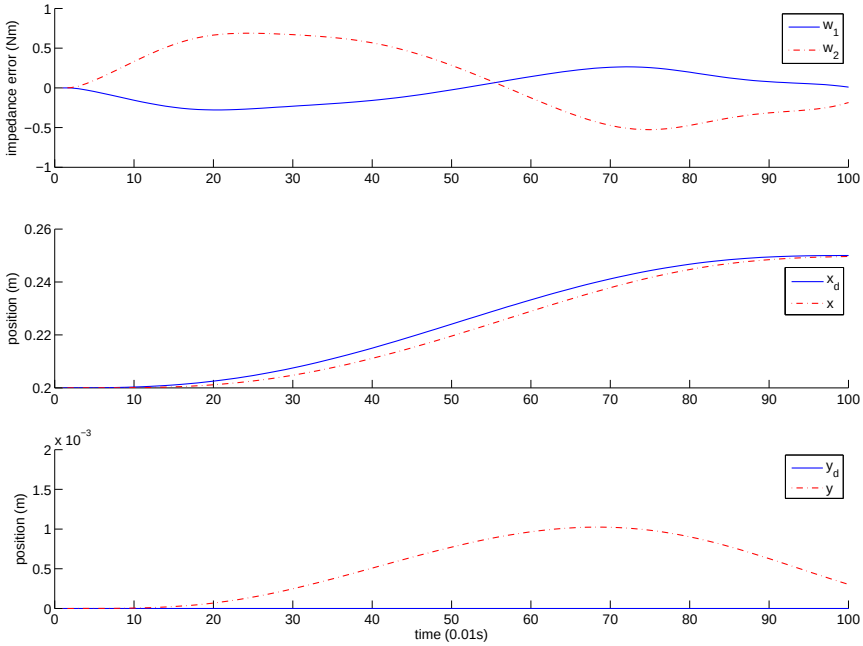



Figure 2: The first case: impedance error, actual trajectory and desired trajectory at k=1

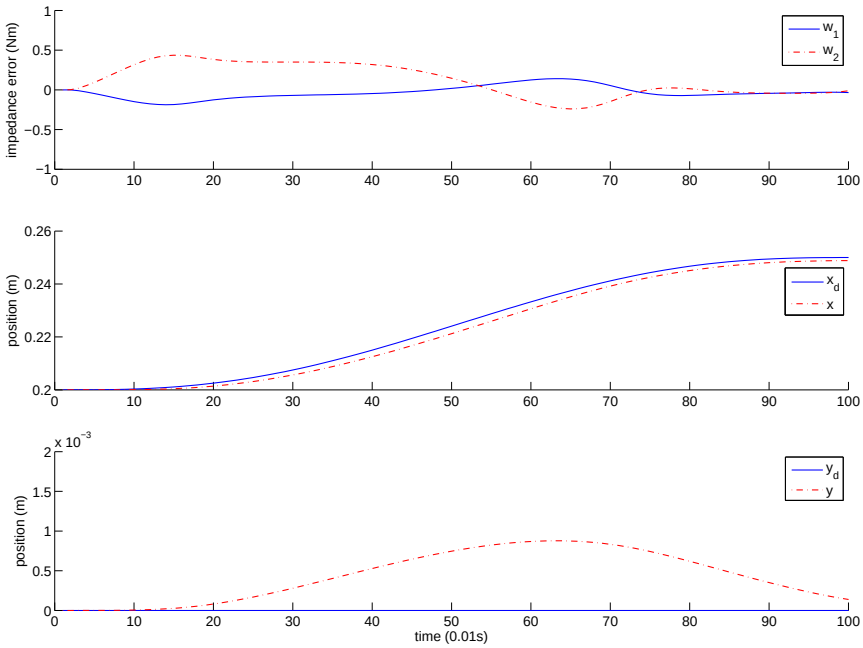



Figure 3: The first case: impedance error, actual trajectory and desired trajectory at k=10

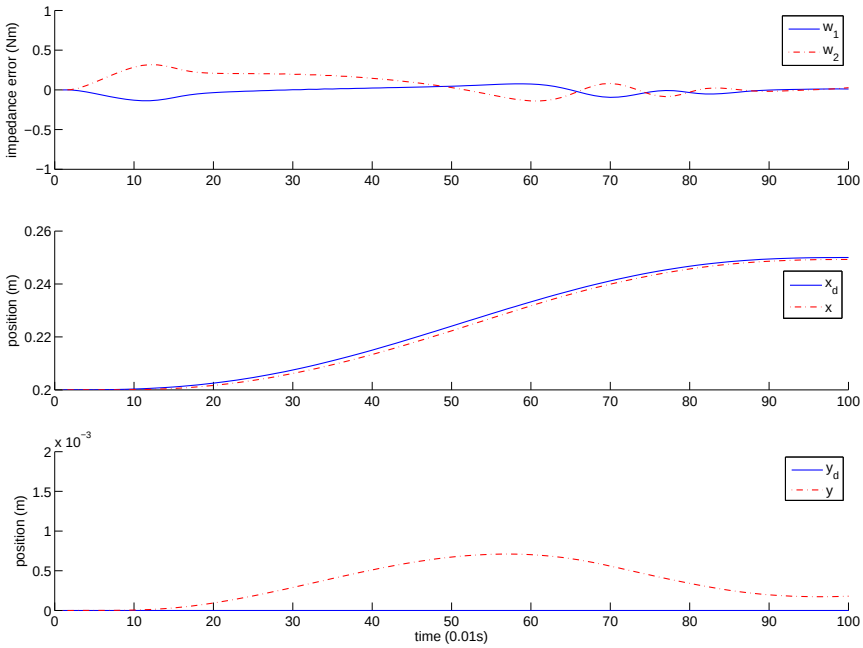



Figure 4: The first case: impedance error, actual trajectory and desired trajectory at k=30

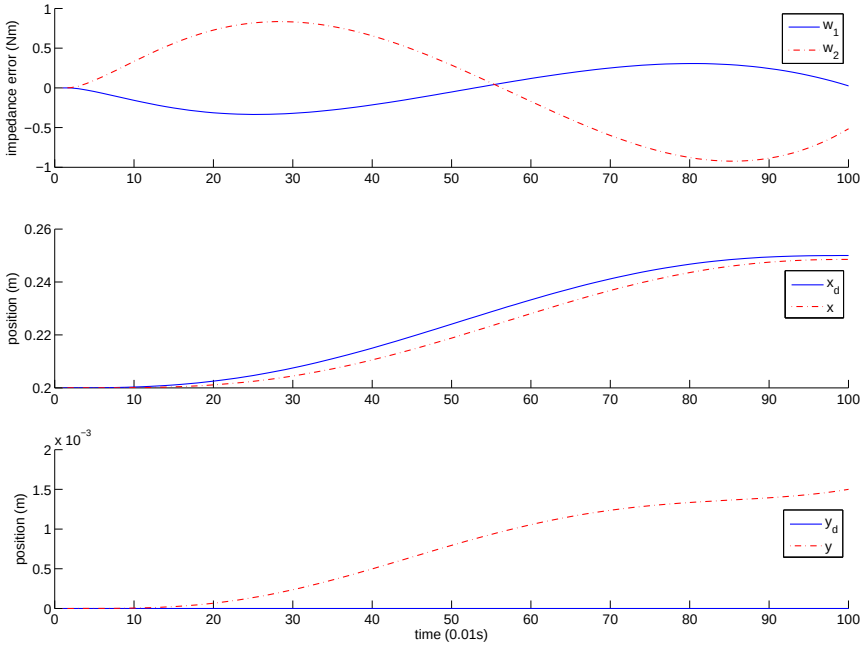



Figure 5: The second case: impedance error, actual trajectory and desired trajectory at k=1

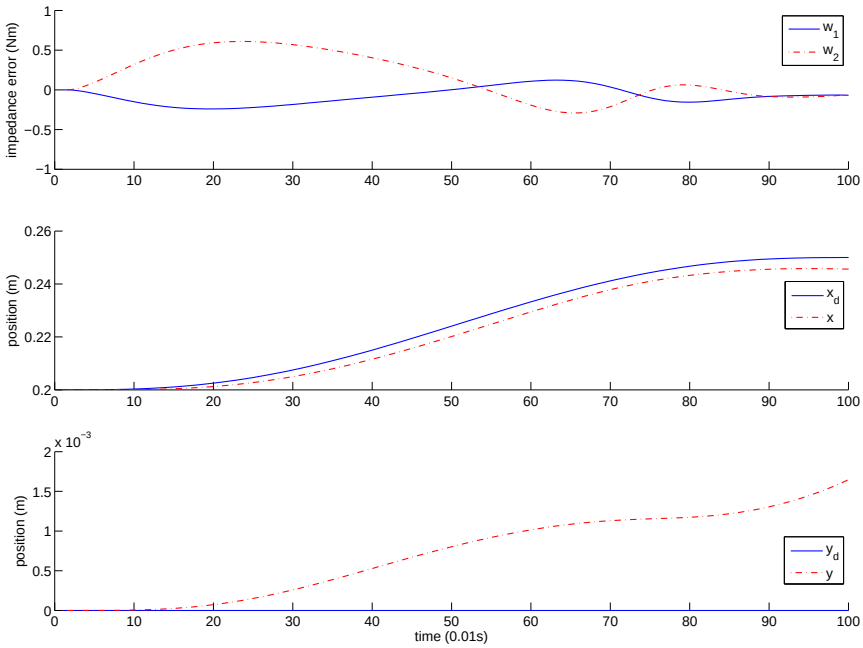



Figure 6: The second case: impedance error, actual trajectory and desired trajectory at k=10

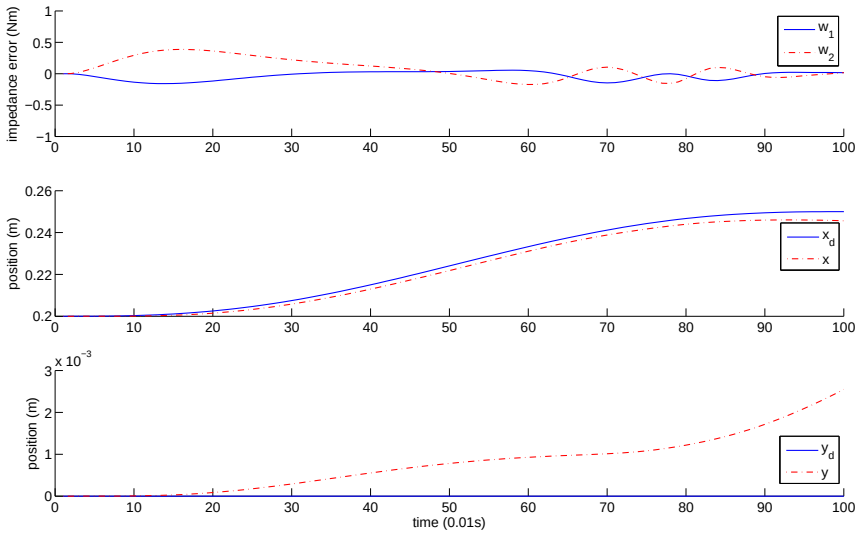



Figure 7: The second case: impedance error, actual trajectory and desired trajectory at k=30

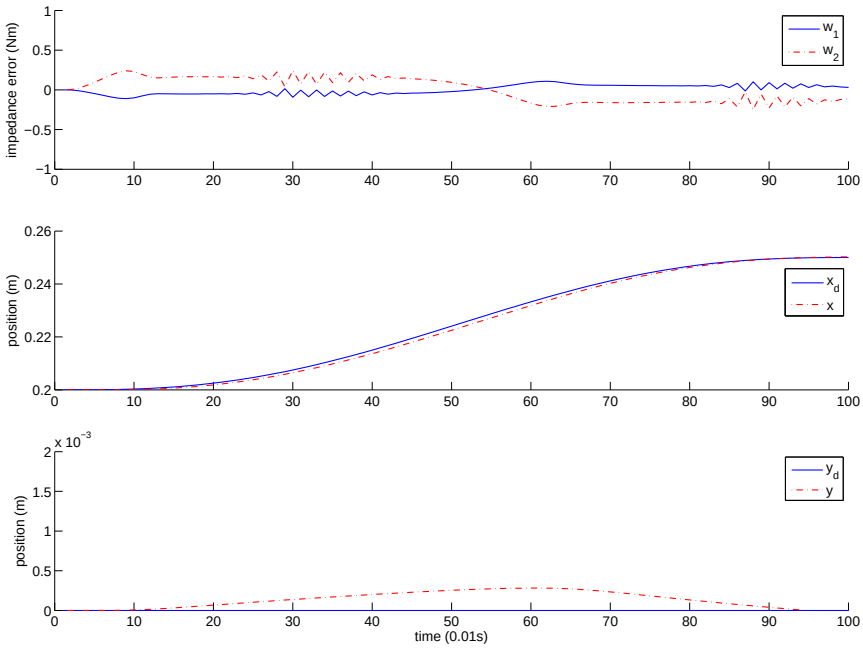



Figure 8: The first case: impedance error, actual trajectory and desired trajectory at k=30 with the method in [16]

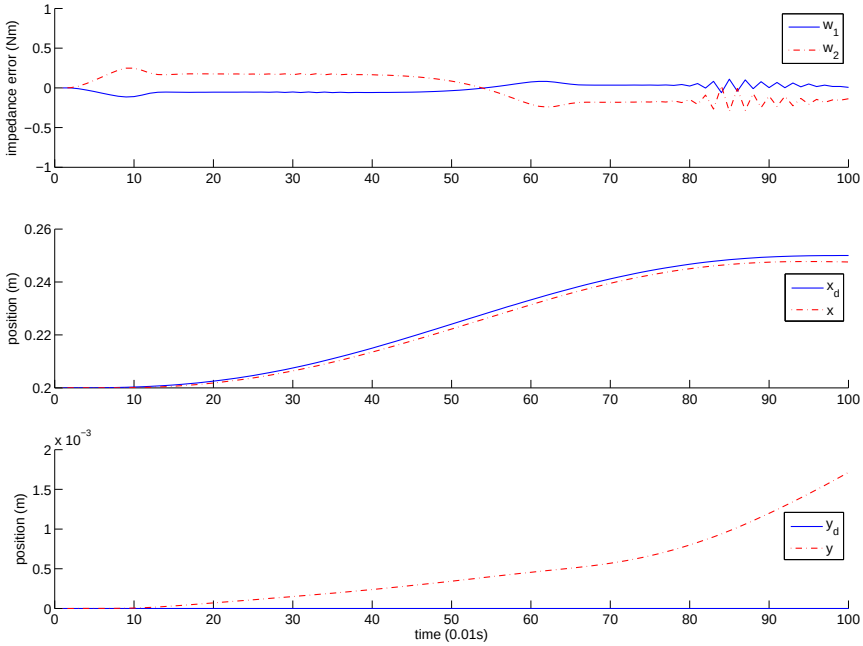



Figure 9: The second case: impedance error, actual trajectory and desired trajectory at k=30 with the method in [16]